Advanced Java Lab

Week-3

Roll Number: 238W1A12C4

1. Implement batch update using JDBC.

Code:

```
import java.sql.*;


public class Mavenproject2 {
   public static void main(String[] args) {


      String url = "jdbc:mysql://localhost:3306/mydb";
      String user = "root";
      String password = "";


      try {
         Class.forName("com.mysql.cj.jdbc.Driver");
         Connection con = DriverManager.getConnection(url, user, password);


         con.setAutoCommit(false); // REQUIRED FOR BATCH


         String sql = "INSERT INTO Person (PersonID, FirstName, City) VALUES (?, ?,
?)";
         PreparedStatement ps = con.prepareStatement(sql);


         // Record 1
         ps.setInt(1, 201);
         ps.setString(2, "Rohit");
         ps.setString(3, "Nagpur");
```

```
            ps.addBatch();

            // Record 2
            ps.setInt(1, 202);
            ps.setString(2, "Sneha");
            ps.setString(3, "Kolkata");
            ps.addBatch();

            // Record 3
            ps.setInt(1, 203);
            ps.setString(2, "Kavya");
            ps.setString(3, "Goa");
            ps.addBatch();

            // Execute all queries in one go
            int[] results = ps.executeBatch();

            con.commit();
            con.close();

            System.out.println("Batch Update Completed!");
            System.out.println("Rows affected: " + results.length);

        } catch (Exception e) {
            System.out.println("Error: " + e);
        }
    }
}
```

Output:

```
--- exec:3.1.0:exec (default-cli) @ mavenproject2 ---
Batch Update Completed!
Rows affected: 3
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  4.982 s
Finished at: 2025-11-22T17:49:25+05:30
------------------------------------------------------------------------
```

2. Create a CRUD application using JDBC with transaction

Code:

```
package com.mycompany.mavenproject2;


import java.sql.*;


public class Mavenproject2 {
   public static void main(String[] args) {


       String url = "jdbc:mysql://localhost:3306/mydb";
       String user = "root";
       String password = "";


       try {
          Class.forName("com.mysql.cj.jdbc.Driver");
          Connection con = DriverManager.getConnection(url, user, password);


          con.setAutoCommit(false); // IMPORTANT


          // INSERT ----------------------------------------------------
```

```
            String insertSQL = "INSERT INTO Person (PersonID, FirstName, City) VALUES
(?, ?, ?)";
            PreparedStatement insert = con.prepareStatement(insertSQL);
            insert.setInt(1, 301);
            insert.setString(2, "Anita");
            insert.setString(3, "Surat");
            insert.executeUpdate();

            // UPDATE ----------------------------------------------------
            String updateSQL = "UPDATE Person SET City=? WHERE PersonID=?";
            PreparedStatement update = con.prepareStatement(updateSQL);
            update.setString(1, "Vadodara");
            update.setInt(2, 301);
            update.executeUpdate();

            // DELETE ----------------------------------------------------
            String deleteSQL = "DELETE FROM Person WHERE PersonID=?";
            PreparedStatement delete = con.prepareStatement(deleteSQL);
            delete.setInt(1, 202); // delete existing user
            delete.executeUpdate();

            // If everything successful → COMMIT
            con.commit();
            System.out.println("Transaction Successful!");

            con.close();
        }
        catch (SQLException e) {
            System.out.println("Database Error: " + e.getMessage());
```

```
            try {
                // ROLLBACK on failure
                Connection con = DriverManager.getConnection(url, user, password);
                con.rollback();
                System.out.println("Transaction Rolled Back!");
            } catch (SQLException ex) {
                System.out.println("Rollback Failed: " + ex.getMessage());
            }
        }
        catch (Exception ex) {
            System.out.println("Error: " + ex.getMessage());
        }
    }
}
```

Output: